

Notions de base - Variables

- ✓ Une variable est un emplacement de mémoire volatile nommé qui sert à stocker / conserver des données.
- ✓ Les variables peuvent être considérées comme des emplacements en mémoire qui contiendront les données lorsque les valeurs sont entrées ou calculées pendant l'exécution du programme.
- ✓ Les valeurs stockées dans les variables peuvent ensuite être modifiées lors de l'exécution du programme.
- ✓ Bien que ce concept puisse sembler nouveau, vous avez déjà vu et traité des variables en algèbre. Vous avez certainement vu quelque chose comme (où x est la variable):

$$3x + 5 = 20$$

$$x = 5$$

Basics – Variables

Une variable peut être vue comme un objet avec les attributs suivants:

1. Nom - une balise utilisée pour identifier la variable dans le code VB.
2. Adresse - un emplacement en mémoire où la valeur des variables est stockée.
3. Type - le "type" de données que la variable peut stocker. Le type de données détermine la plage possible de valeurs que la variable peut stocker, ainsi que l'espace en mémoire que la variable occupera. Le type déterminera également les opérations qui peuvent être effectuées sur (ou vers) la variable.
4. Valeur - contenu de la mémoire commençant à l'adresse et ayant la taille déterminée par le type.
5. Scope / Visibilité - où dans un programme une valeur de variable est visible, c'est-à-dire où elle peut être référencée.
6. Durée de vie - lorsqu'un programme a une valeur de variable visible, c'est-à-dire quand il peut être référencé.

Basics – Variables

- ✓ Toutes les variables dans VB doivent être déclarées avant d'être utilisées.
- ✓ L'instruction de déclaration est l'endroit où le programmeur attribue un nom à la variable ainsi qu'un type associé.
- ✓ En Visual Basic, l'instruction de déclaration de variable a le général:

```
access_type var_name As type [= initial_value]
```



Les bases – Types de données

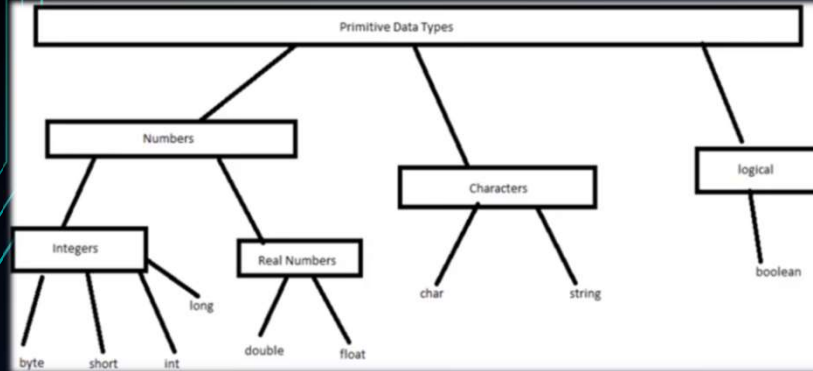
Numérique

- ✓ Byte
- ✓ Integer
- ✓ Long
- ✓ Double
- ✓ Single
- ✓ Decimal

Non-Numérique

- ✓ String
- ✓ Date
- ✓ Boolean
- ✓ Char

Les bases – Types de données



Principes de base - Règles de dénomination

Le nom de la variable doit suivre plusieurs règles, car il:

1. Ne peut utiliser que des lettres, des chiffres et des traits de soulignement (_);
2. Doit commencer par une lettre ou un trait de soulignement;
3. Ne peut pas avoir d'espaces;
4. Impossible d'utiliser des caractères spéciaux;
5. Ne peut pas être un mot clé.

Principes de base - Règles de dénomination

Les conventions de dénomination permettront de reconnaître plus facilement les variables, leur type et ce qu'elles contiennent.

Les variables doivent:

1. Ayez un nom significatif.
2. Commencez par une lettre minuscule.
3. Incluez un préfixe qui décrit le type de données.
4. Incluez un nom dans lequel la première lettre de chaque mot est en majuscule.

Type	Prefix
Single	sgl
Double	dbl
Integer	int
Long	lng
String	str
Boolean	bln

Bases - Conventions de dénomination

```

Dim strLastName As String ' User's last name
Dim intAge As Integer ' User's age
Dim dblHeight As Double ' User's height in cm

Dim blnVote As Boolean ' True if user is old
                        ' enough to vote;
                        ' false otherwise

intAge = 17.4 ' ok, but doesn't work as
              ' expected. intAge stores 17

dblHeight = 175.36 ' dblHeight stores 175.36
  
```

Basics – Creation d'une variable

Typiquement en VB, le type d'accès d'une variable est spécifié à l'aide du mot-clé « Dim ». En affectant un nom de variable, un type et un type d'accès, nous observons les déclarations de déclaration de variable suivantes:

```
Dim intAge As Integer
Dim intSum As Integer
Dim strFirstName As String
Dim intAge, intSum As Integer
```

Notions de base - Types de données - Plages et préfixes

Data Types, Ranges, and Prefixes

Data Type	Prefix	Range	Bytes
Byte	byt	0 to 255	1
Short	sho	-32,768 to 32,767	2
Integer	int	-2,147,483,648 to 2,147,483,647	4
Long	lng	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807	8
String	str	0 to ~2 billion	1 per character
Decimal	dec	1.0e-28 to 7.9E+28	16
Single	sng	1.5E45 to 3.4E+38	4
Double	dbl	5.0E324 to 1.7E+308	8
Date	dat	January 1, 0001 to December 31, 9999 0:00:00 to 23:59:59	8
Boolean	bln	True or False	2
Char	chr	one Unicode character	2
Object	obj	data of any type	—

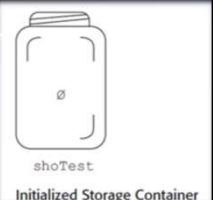
Bases - Types de données - Déclaration

- ✓ La plupart des variables avec lesquelles vous travaillerez sont numériques afin que vous puissiez gérer les nombres.
- ✓ Un autre type de données important est une chaîne. Il est utilisé pour stocker une série de personnages.
- ✓ Utilisez une chaîne pour stocker votre nom, une adresse ou tout ce qui contient du texte.
- ✓ Il est également utilisé pour les nombres qui ne sont pas traités comme des nombres. Le numéro de sécurité sociale, le code postal ou la carte d'étudiant sont tous de bons exemples de numéros qui ne sont pas vraiment des «numéros».

```
Dim strLastName As String
```

Bases - Types de données - Initialisation

- ✓ Avant d'utiliser une variable, elle doit être initialisée.
- ✓ L'initialisation variable signifie la définition de la valeur initiale pour elle.
- ✓ Vous pouvez initialiser une variable en même temps que vous la déclarez.
- ✓ Il existe des initialisations par défaut pour certains types de données:
 - ✓ Numérique: zéro - 0
 - ✓ Chaîne: nulle ou chaîne vide ("")



```
Dim <name> As <Datatype> [= <data>]
```

```
Dim numOfCows As Integer
```

```
Dim temperature As Single
```

```
Dim sentence As String = "Hello everyone!"
```

```
Dim strLastName As String
```

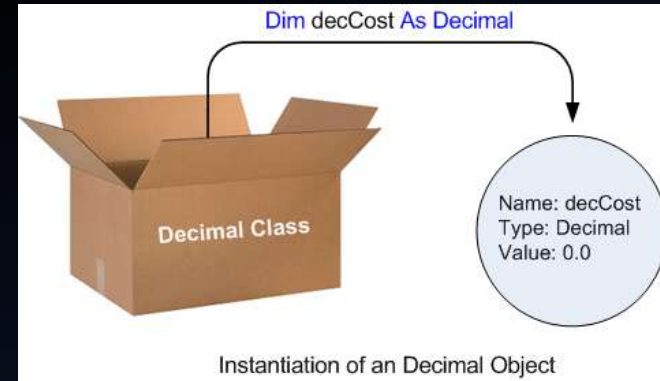
```
Dim shoCaseSize As Short = 24
```

```
Dim strLastName As String = "Jefferson"
```

Types de données – Valeur par défaut

- ✓ Lors de la déclaration de variables, si aucune valeur initiale n'est affectée, VB initialise toutes les variables à zéro par défaut, ou une valeur qui ressemble étroitement à zéro.
- ✓ Ainsi, les types entiers obtiennent une valeur par défaut de 0, les types doubles obtiennent une valeur par défaut de 0.0 et les types chaîne obtiennent une valeur par défaut de rien (Nothing).
- ✓ En ce qui concerne les types de chaînes, il est important de noter que rien, la chaîne vide (""), et une chaîne contenant un espace (" ") sont des objets très différents:
 - ✓ Rien n'a pas de longueur, car ce n'est pas un objet existant (c'est une référence NULL).
 - ✓ La chaîne vide (""), est un objet, avec essentiellement rien en elle, a donc la longueur de zéro.
 - ✓ Enfin, une chaîne contenant un espace (" ") a une longueur d'un.

Types de données – Valeur par défaut



Conseils

Codes postaux: vous ne pouvez pas enregistrer les codes postaux sous forme de nombre. Bien que la plupart d'entre eux fonctionnent comme un nombre, vous ne faites jamais de calcul avec eux, il n'est donc pas nécessaire de les stocker sous forme de nombres. En outre, les codes postaux des États de la Nouvelle-Angleterre commencent tous par zéro et les variables numériques ne peuvent pas afficher les zéros en tête.

Une chaîne ne peut pas contenir de guillemets. Les guillemets sont utilisés pour indiquer où la chaîne commence et se termine. Lorsque vous placez un guillemet au milieu d'une chaîne, l'ordinateur pense que c'est la fin de la chaîne. Si vous avez vraiment besoin de guillemets autour de quelque chose dans une chaîne - un titre, par exemple - vous devez utiliser des guillemets simples (').

Bases - Affectation

- ✓ L'affectation de valeur est un moyen de mettre une valeur dans le conteneur de stockage (variable).
- ✓ La variable est toujours à gauche et la valeur placée dans la variable est toujours à droite.
- ✓ Utilisez le signe égal pour attribuer la valeur.

Méthodes d'affectation

Affectation directe

shoTestScore = 88

Variable à variable

strNewAddress = strOldAddress

Calcul mathématique

shoTotal = shoNum1 + shoNum2

Bases – Opérateurs mathématiques

Mathematical Operators in Visual Basic

Operation	Mathematical Sign	VB Equivalent
Addition	+	+
	A + B	shoNum1 + shoNum2
Subtraction	–	–
	X – Y	decSubtotal – decDiscountAmount
Multiplication	×	*
	R * S	decSubtotal * decDiscountRate
Division	÷	/, \, Mod
	H ÷ K	shoTotal / shoNumScores shoNumCases \ shoCaseSize shoTotalBottles Mod shoCases
Exponentiation	() ⁿ T ²	^ shoSide^3

Bases - Règles de priorité des opérateurs mathématiques

L'ordinateur suit plusieurs règles de priorité qui sont presque les mêmes que les règles de mathématiques. Dans VB, faites:

- ✓ Qu'y a-t-il entre parenthèses en premier et que vous sortez,
- ✓ Exponentiation avant tout calcul mathématique,
- ✓ Multiplication et division de gauche à droite,
- ✓ Puis addition et soustraction de gauche à droite.

Bases – Constantes

- ✓ Les constantes sont des variables ayant des valeurs fixes.
- ✓ Les constantes sont très utiles, et faciles à écrire.
- ✓ Les constantes sont créées, nommées et affectées d'une valeur tout en une seule instruction.
- ✓ Utilisez le mot-clé «Const» pour créer une constante, comme suivant: `Const sngSalesTaxRate As Single = 0.05`
- ✓ Une constante est définie lorsqu'elle est déclarée et ne peut pas être modifiée accidentellement dans le reste du code.

```
Const strTitle As String = "A Tale of Two Cities"
```

Bases – Constantes

- ✓ Il est recommandé d'utiliser des constantes chaque fois que vous avez des valeurs qui ne changeront pas pendant l'exécution d'un programme.

Bases – Exercise**VB Quiz**

What is wrong with each of the following?

```
shoNum + shoNum2 = shoTotal
```

```
shoNum = 10 * 4
```

```
Dim decPrice As Decimal = "$4.99"
```

```
shoCube = (shoSide)3
```